

# 複数コンテキストドメインにまたがる Linked Data を用いた コンテキストモデル構築方法の提案と評価

M2014SE007 内海 太祐

指導教員 青山 幹雄

## 1. はじめに

コンテキストウェアシステムでは、複数のコンテキストドメインを統合したコンテキストモデルを扱う必要がある。しかし、異なるコンテキストドメインにおいて、共通述語が存在しない場合のコンテキストモデル構築方法は確立されていない。

## 2. 研究課題

### (1) リソース表記が異なる場合の統合

異なるコンテキストドメインでは、リソース表記が異なる場合がある。リソース表記の文字列による探索、関係付けは、リソース表記が異なる場合は対象とするリソースの発見が困難であり、意味合いが異なる同一の表記のリソースに誤った関係付けを行ってしまう可能性がある。

### (2) 共通述語が存在しない場合の統合

共通述語が存在しない場合においては、関係付けに必要な情報を得ることができず、コンテキストドメインの統合が困難である。

## 3. 関連研究

### 3.1. Linked Data

Linked Data は RDF によって記述され、データを Web 上で相互にリンクする方法である[2]。Linked Data への問合せ言語として SPARQL がある。

### 3.2. Linked Data を用いた情報推薦

観光に関する複数コンテキストドメインにまたがるコンテキストモデルを構築する方法が提案されているが、任意のコンテキストドメインの統合は実現されていない[4]。

### 3.3. 属性情報を用いた関係付け

リソースの属性情報を参照することによって表記揺れなど、曖昧性のあるリソースが存在するコンテキストドメインを統合することが可能である[5]。

### 3.4. Silk

Silk は異なるコンテキストドメイン間のリソースのリンク生成を支援するフレームワークである。対象とする2つのコンテキストドメインのリソースを主語とする述語、目的語の類似性メトリクスをもとに関係性を判定している。属性情報が少ない場合や共通述語が存在しない場合はリンク生成を行うことができない[3]。

## 4. アプローチ

### 4.1. コンテキストドメインの統合

分散しているコンテキストドメインのリソースを他のコンテキストドメインのリソースに対して、RDF の述語による関係付けを行うことをコンテキストドメインの統合と定義する。

### 4.2. アプローチの全体像

異なるコンテキストドメインの統合において、リソース表記文字列の比較のみでは誤った関係付けを行ってしまう可能性がある。また、共通述語が存在しない場合、属性情報による関係付けが困難である。本研究では、統合する2つのコンテキストドメインから共通述語を抽出し、リソースの属性情報による関係付けと外部 LOD(Linked Open Data)を用いた関係付けを行う。

## 5. 提案方法

### 5.1. 提案プロセス

提案プロセスの詳細を以下に示す。

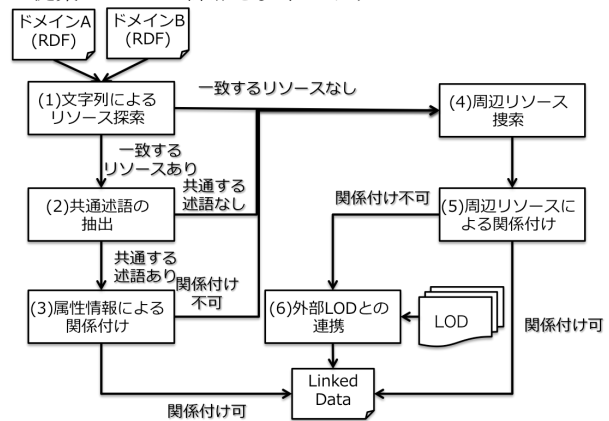


図 1 提案プロセス

### 5.2. 提案プロセスの詳細

提案プロセスの詳細を示す。

#### (1) 文字列によるリソース探索

統合する対象とするリソースを決定するために、文字列による探索を行う。統合の対象とするリソースを統合元コンテキストドメインの RDF から特定の述語を持つリソースを指定し、SPARQL クエリを実行することによって取得する。

#### (2) 共通述語の抽出

(1)のプロセスにおいて、対象となるリソースが存在するならば、その2つのリソースの共通述語の集合を、SPARQL クエリを実行することによって抽出する。

#### (3) 属性情報による関係付け

(2)のプロセスにおいて、共通述語が存在する場合は、属性情報を参照し関係付けを行う。

#### (4) 周辺リソース探索

対象とするリソースの主語や目的語、述語の集合を周辺リソースとする(図 2)。SPARQL クエリを実行することによって、対象とするリソースの周辺リソースを取得する。

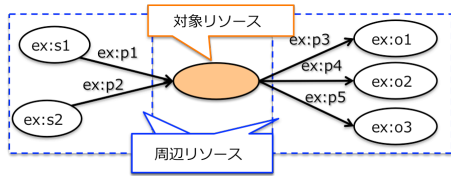


図 2 周辺リソースの定義

- (5) 周辺リソースによる関係付け  
周辺リソースを参照してリソースの関係付けを行う(図 3).

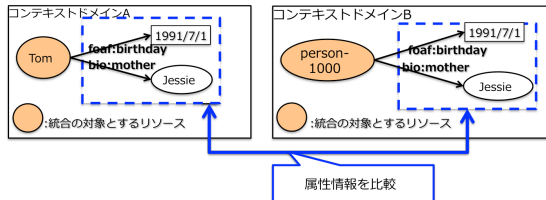


図 3 周辺リソースによる関係付け

また、周辺リソースの属性情報を比較することによって関係付けを行う(図 4).

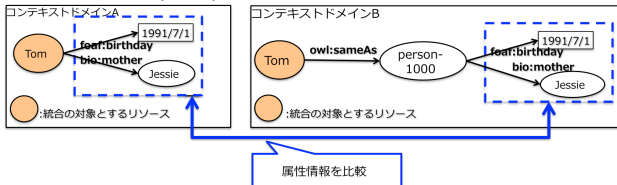


図 4 周辺リソースの属性情報による関係付け

- (6) 外部 LOD との連携  
周辺リソースが少ないコンテキストドメイン間の統合や共通述語が存在しないコンテキストドメイン間の統合を行う場合は統合を行うことができない。そのため DBpedia といった、多くの情報が付加されている LOD の属性情報を参照して関係付けを行う。外部 LOD との連携は対象とする2つのコンテキストドメインについて行う。

外部 LOD との連携は以下のプロセスからなる。

- 文字列によるリソース探索
- 共通述語の抽出
- 属性情報による関係付け
- 周辺リソース探索
- 周辺リソースによる関係付け

## 6. プロトタイプの実装

提案方法の評価を行うためにプロトタイプを実装した。プロトタイプの実行環境を表 1 に示す。

表 1 プロトタイプの実行環境

OS	Mac OS 10.9.3
CPU	Core i5 1.4GHz
メモリ	8GB

関係付けを自動で行うために、提案方法を Java で実装した。表 2 にその詳細を示す。

表 2 プロトタイプの構成

Java 実行環境	JRE_1.8.0_05
SPARQL API	Apache Jena 3.0.1
RDF ストア	Apache Jena Fuseki 2.3.1
ステップ数	401

## 7. 例題への適用

### 7.1. 適用する Linked Data

提案方法を適用する Linked Data として、観光情報ドメイン、イベント情報ドメイン、ロコミ情報ドメイン、天気情報ドメインの 4 つのコンテキストドメインを統合する。外部 LOD として DBpedia Japanese を使用する[1]。提案方法を適用する Linked Data の構造を図 5 にそれぞれのコンテキストドメインのトリプル数を表 3 に示す。例題では、1 つ以上の属性情報が一致した場合に owl:sameAs による関係付けを行った。

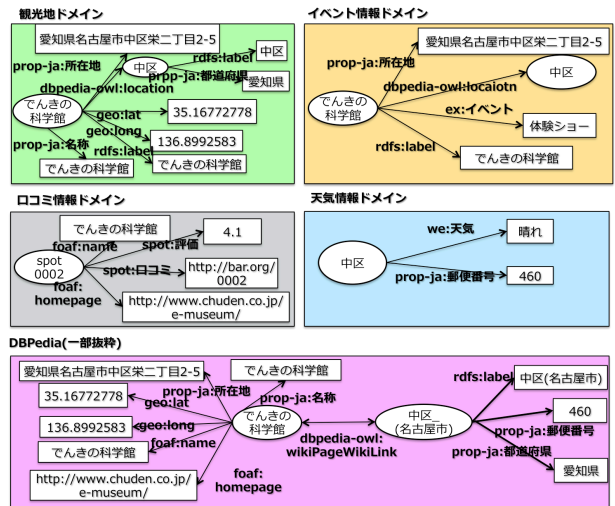


図 5 適用する Linked Data

表 3 各コンテキストドメインのトリプル数

観光情報ドメイン	680
イベント情報ドメイン	485
天気情報ドメイン	140
ロコミ情報ドメイン	388

### 7.2. 適用プロセス

- (1) リソース表記が同じで、共通述語が存在する場合  
観光スポットをキーとして関係付けを行う。図 5 に示すように観光地ドメインとイベント情報ドメインはリソース表記が同じであり、リソース”でんきの科学館”には”prop-ja:所在地”や”dbpedia-owl:location”といった述語が存在するため、属性情報による関係付けを行い、コンテキストドメインを統合する。

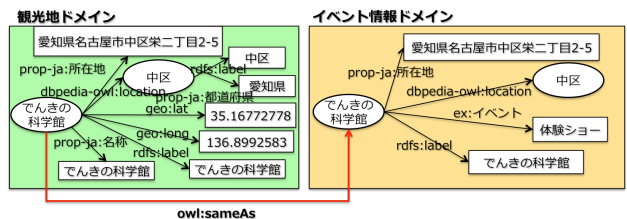


図 6 リソース表記が同じで共通述語が存在する場合の適用結果

- (2) リソース表記が同じで、共通述語が存在しない場合  
観光地ドメインと天気情報ドメインは地名をキーとして関係付けを行う。図 5 に示すように、リソース”中区”には共通述語が存在しない。そのため、周辺リソースを取得し、周辺リソースによる関係付けを行うが比較可能なリソースが存在しないため、関係付けできない。関係付けできなかったリソースに対して外部 LOD との連携を行う。

外部 LOD に対して文字列によるリソース探索を行った結果、全てのリソースに対して統合の対象となるリソースが発見できた。そのリソースに対して属性情報による関係付けを行う。その結果を表 4 に示す。

表 4 属性情報による関係付け結果

ドメイン名	関係付け可	関係付け不可
観光地ドメイン	45	7
天気情報ドメイン	63	7

表 4 に示すように 7 個のリソースについて関係付けを行うことができなかった。関係付けできなかったリソースは”中区”や”東区”といった全国に同じ地名が存在するリソースであった。関係付けできなかった 7 個のリソースについて周辺リソースによる関係付けを行う。図 12 に示すように、周辺リソースの属性情報を参照することによって関係付けを行う。

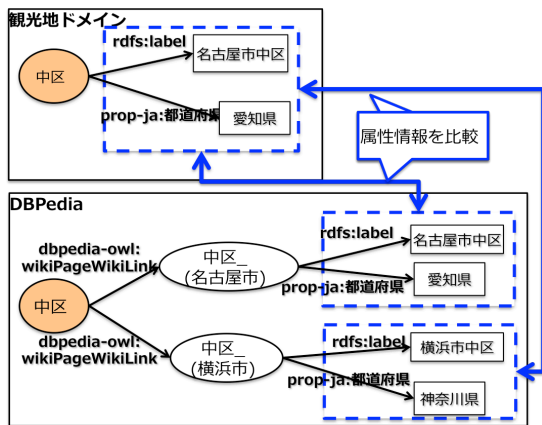


図 7 周辺リソースの属性情報による関係付け

図 8 に観光地ドメインと天気情報ドメインに提案方法を適用した結果を示す。

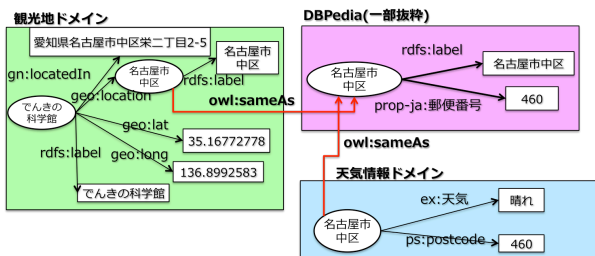


図 8 リソースの表記が同じ共通述語が存在しない場合の適用結果

(3) リソース表記が異なり共通述語が存在しない場合

観光地ドメインとロコミ情報ドメインを、観光スポットをキーとして関係付けを行う。図 5 に示すように、リソースの表記が異なり、共通述語が存在していない。そのため、周辺リソースを取得し関係付けを行う。しかし、関係付けに必要な情報が存在しない。統合の対象となる 98 個のリソースすべてが周辺リソースによる関係付けが不可能であるため、外部 LOD との連携を行う。表 5 に観光地ドメインとロコミ情報ドメインと DBpedia との関係付けを行った結果を示す。

表 5 DBpedia との関係付け結果

ドメイン名	関係付け可	関係付け不可
観光地ドメイン	42	56
ロコミ情報ドメイン	36	62

図 9 に観光地ドメインと外部 LOD との連携におけるプロセスの流れを示す。

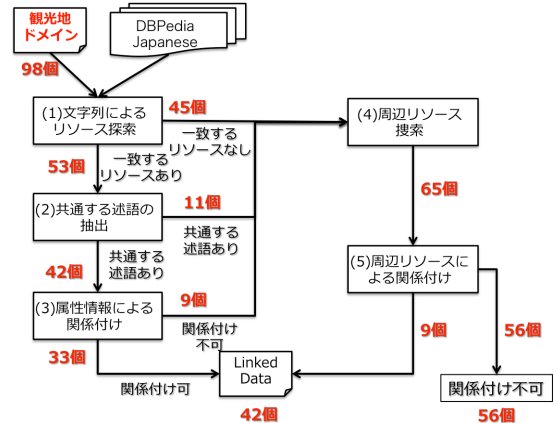


図 9 観光地ドメインと DBpedia の適用プロセス詳細

図 10 に DBpedia を介して観光地ドメインとロコミ情報ドメインに提案方法を適用した結果を示す。

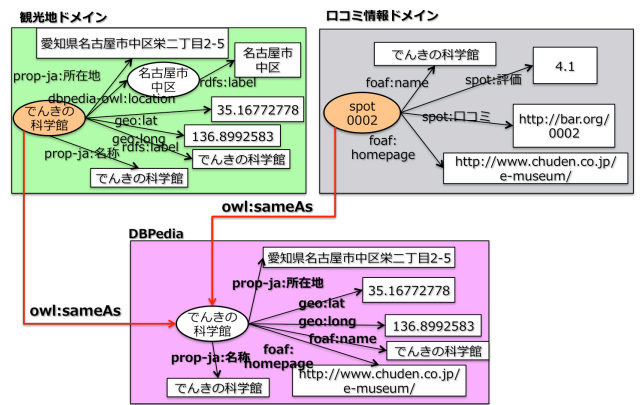


図 10 リソース表記が異なり共通述語が存在しない場合の適用結果

7.3. 適用結果

提案方法を例題に適用した結果を図 11 に示す。

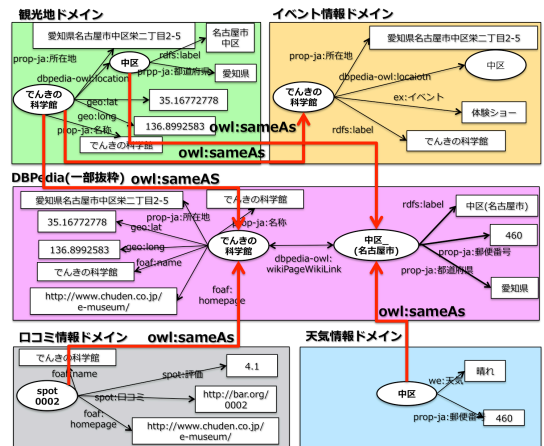


図 11 提案プロセスを適用した結果

それぞれの場合についての適用結果を示す。

- (1) リソース表記が同じで、共通述語が存在する場合  
属性情報の比較による関係付けで全てのリソースを統合することが可能であった。
- (2) リソース表記が同じで、共通述語が存在しない場合  
比較可能な属性情報や周辺リソースによる関係付けができず、外部 LOD との連携を行った。LOD に存在するリソースの属性情報による関係付けによって、7 個を除くリソースについて関係付けを行うことができた。関係付けできな

った 7 個のリソースについては周辺リソースを参照することによって関係付けを行うことができた。

(3) リソース表記が同じで、共通述語が存在しない場合

外部 LOD を介して統合できたリソース数は、観光地ドメインが 98 個中 42 個、ロコミ情報ドメインが 98 個中 36 個であった。それ以外にリソースについては、DBPedia にリソースが存在しない、またはリソースは存在するが比較可能な属性情報が存在しないために関係付けできなかった。図 12 に観光地ドメインと DBPedia の関係付け可否の内訳を示す。

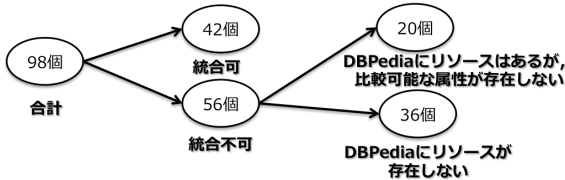


図 12 観光地ドメインと DBPedia の関係付け可否内訳

## 8. 評価と考察

### 8.1. 例題に対する評価

4 つの異なる特性を持つコンテキストドメインの統合を例題として扱った。以下にその評価を示す。

(1) リソースの表記が同じで共通述語が存在する場合

文字列をキーとした探索によって統合する候補となるリソースを発見した。そのリソースの属性情報を比較することによって関係付けを行うことができ、属性情報の比較による関係付けは有意である。

(2) リソースの表記が同じで共通述語が存在しない場合

外部 LOD を介した関係付けを行った結果、すべての対象リソースに対して関係付けを行うことが可能となった。“中区”や“東区”といった全国に複数あり、一意に決定できないリソースに関しても、同一リソースを意味する owl:sameAs や DBPedia の dbpedia-owl:wikiPageRedirects のリンクタイプで関係付けられた周辺リソースを探索、比較によって一意に決定することが可能となった。

(3) リソースの表記が異なり共通述語が存在しない場合

リソースの探索では、関係付けの対象となるリソースを発見することができず、周辺リソースによる関係付けも不可能であるため、外部 LOD による関係付けを行った。外部 LOD の周辺リソースを利用することによって、表記揺れがあるリソースであっても関係付けを行うことができた。また、リソースの表記形式が異なるコンテキストドメイン間であっても関係付けを行うことが可能であると考えられる。

### 8.2. 研究課題に対する評価

(1) リソース表記が異なるコンテキストドメインの統合

リソース表記が異なるコンテキストドメインの場合は、属性情報や周辺リソースによる関係付けを行うことによって統合が可能になった。属性情報を比較することによって、意味合いが異なる同一表記のリソースへの誤った関係付けを防止することができ、周辺リソースを探索、参照することによって表記が異なるリソースに対して関係付けを行うことが可能となった。

(2) 共通述語が存在しないコンテキストドメインの統合

共通述語を持たないコンテキストドメインの場合は、外部 LOD から関係付けに必要な情報を得ることによって、共通述語を持たないコンテキストドメインを統合することが可能であった。

### 8.3. 提案方法に対する評価

(1) 属性情報を用いた関係付け

属性情報の比較を行うことによって異なる意味合いの同一表記リソースや意味合いが曖昧なリソースとの誤った関係付けを防止することが可能になったと考える。

(2) 周辺リソースによる関係付け

周辺リソースを参照することによって、リソースの表記揺れに対応することが可能となった。また owl:sameAs や DBPedia における dbpedia-owl:wikiPageDisambiguates, dbpedia-owl:wikiPageRedirects で関係付けられたリソースをたどることによって異なる表記のリソースや意味合いが曖昧なリソースに対して関係付けを行うことが可能になった。

(3) 外部 LOD を用いた関係付け

外部のリソースを使用することによって、共通の述語が持たないコンテキストドメイン間でもリソースの統合が可能であるが、LOD に比較可能な属性情報が存在しない場合は統合を行うことができない。統合の可否は LOD が持つ情報量とリソースの特性に依存する。

## 9. 今後の課題

周辺リソースの属性情報を利用して関係付けを行う場合、属性情報を取得する SPARQL クエリが RDF ストアに与える負荷が大きくなるため、取得する周辺リソースを絞る必要がある。関係付けられたリソースが実際に正しい関係付けを行われたリソースであるかどうかの判断は人間が行わなければならない。その関係付けが正しいかの検証を定量的に行う必要がある。プロトタイプは小規模なデータのコンテキストドメインに対して適用した。大規模なデータを持つコンテキストドメインにスケールするか検証する必要がある。

## 10. まとめ

分散しているコンテキストドメインの情報は、リソース表記が異なったり共通述語が存在せず、それらのリソースを統合するのは困難である。

本研究では、異なる表記のリソースを持つコンテキストドメインを統合するためにリソースの属性情報や周辺リソースを用い、外部 LOD の情報を用いることによって共通述語を持たないコンテキストドメインを統合する方法を提案した。提案方法を例題に適用することによって有用性を確認した。今後の課題として、取得する周辺リソースの絞り込み、リンクの定量的な評価、大規模なデータにスケールするかを検証する必要がある。

### 参考文献

- [1] DBPedia Japanese, <http://ja.dbpedia.org/>.
- [2] T. Heath, et al., *Linked Data: Evolving the Web into a Global Data Space*, Morgan & Claypool, 2011.
- [3] J. Volz, et al., *Discovering and Maintaining Links on the Web of Data*. In Proc. of the 8th International Semantic Web Conference, (ISWC '09), Berlin, Heidelberg, 2009, pp. 650–665.
- [4] 大河原 渉ほか, *Linked Data の構造分析に基づく高速道路立ち寄りサービス*, 人工知能学会論文集 Vol. 28, 2014.
- [5] 山根 昇平ほか, *オープンデータの Linked Data への変換*, セマンティックとオントロジー研究会, SIG-SWO-A1402-03, 2014, pp1-6.